

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION FOR LETTERS PATENT

**BORE-Resistant Digital Goods Configuration and
Distribution Methods And Arrangements**

Inventor(s):

Marcus Peinado

Mariusz H. Jakubowski

Ramarathnam Venkatesan

1 **TECHNICAL FIELD**

2 This invention relates to digital goods and content, and more particularly to
3 Break-Once, Run-Everywhere (BORE) resistant digital goods configuration and
4 distribution methods and arrangements that significantly protect rights associated
5 with the distribution and use of digital goods and digital content.
6

7 **BACKGROUND**

8 Digital goods (e.g., software products and the like) and data or digital
9 content (e.g., music, video, books, etc.) are often distributed to consumers via
10 fixed computer readable media, such as, for example, a compact disc (CD-ROM),
11 digital versatile disc (DVD-ROM), soft magnetic diskette, or hard magnetic disk
12 (e.g., a preloaded hard drive). More recently, consumers have been able to
13 download digital goods and digital content directly to their computers using data
14 communication services, such as, for example, those associated with the Internet.

15 One of the on-going concerns with such distribution techniques, however, is
16 the need to provide digital rights management (DRM) protection to prevent
17 unauthorized distribution, copying and/or illegal operation of, or access to the
18 digital good and content. An ideal digital goods distribution system would
19 substantially prevent unauthorized distribution/use of the digital goods and
20 content.

21 Various DRM techniques have been developed and employed in an attempt
22 to thwart potential software pirates from illegally copying or otherwise distributing
23 the digital goods to others. For example, one DRM technique includes requiring
24 the consumer to insert the original CD-ROM or DVD-ROM for verification prior
25 to enabling the operation of a related copy of the digital good. Unfortunately, this

1 DRM technique typically places an unwelcome burden on the honest consumer,
2 especially those concerned with speed and productivity. Moreover, such
3 techniques are impracticable for digital goods that are site licensed, such as, for
4 example, software products that are licensed for use by several computers, and/or
5 for digital goods that are downloaded directly to a computer. Additionally, it is not
6 overly difficult for unscrupulous individuals/organizations to produce working
7 pirated copies of the CD-ROM, for example.

8 Another DRM technique includes requiring or otherwise encouraging the
9 consumer to register the digital good with the provider, for example, either through
10 the mail or online via the Internet or a direct connection. Thus, the digital good
11 may require the consumer to enter a registration code before allowing the digital
12 good to be fully operational or the digital content to be fully accessed.
13 Unfortunately, such DRM techniques are not always effective since unscrupulous
14 individuals/organizations need only break through or otherwise undermine the
15 DRM protections in a single copy of the digital good. Once broken, copies of the
16 digital good can be illegally distributed, hence such DRM techniques are
17 considered to be Break-Once, Run-Everywhere (BORE) susceptible.

18 Consequently, there is need for digital goods configuration and/or
19 distribution methods and arrangements that are significantly more BORE-resistant.
20 Preferably, the BORE-resistant methods and arrangements will be easy to
21 implement and cost effective for the digital good developer and/or the content
22 producer, supportive of online distribution and multiple station licensing,
23 traceable, difficult to undermine, and not overly burdensome on the consumer.
24
25

1
2 **SUMMARY**

3 The present invention provides DRM (Digital Rights Management)
4 software, distribution methods, and arrangements that are designed to protect
5 software, content (e.g., music, video, books, etc.), and other digital goods
6 (hereinafter, "digital goods" refers to all the above). The DRM software is
7 configured to be resistant to Break Once, Run Everywhere (BORE) attacks. The
8 BORE-resistant methods and arrangements are easy and cost effective for the
9 digital good developer or content producer to implement, and are not overly
10 burdensome on the consumer. The various methods and arrangements support
11 traditional and online distribution techniques, and are adaptable for site licensing.
12 The resulting digital good is substantially difficult to undermine on any significant
13 scale, because each copy is uniquely configured for use by an authorized
14 consumer/computer.

15 Thus, for example, in accordance with certain aspects of the present
16 invention, improved DRM security is provided by individualizing the digital good
17 for each consumer using selective program flow manipulation techniques. The
18 program-flow-manipulation techniques are combined with encryption and/or
19 cryptography keying techniques or other unique/trusted identifying techniques to
20 individualize the configuration of a digital good for each authorized consumer.

21 The digital good can be distributed in one or more parts that are selectively
22 modified and/or otherwise provided to an authorized consumer having the
23 applicable security keys and/or other unique/trusted identifier information needed
24 to complete the configuration of an individualized and operatively unique
25 modified digital good.

1 The modified digital good is unique for each consumer/computer, because
2 the security keys and/or other unique/trusted identifiers are used as inputs during
3 program flow manipulation within the source's/consumer's computer. Subsequent
4 initialization/operation of the uniquely configured modified digital good can
5 include verifying the presence of certain consumer/computer identifying data to
6 further promote DRM protection. Consequently, the modified digital good and the
7 distribution techniques are substantially less susceptible to BORE tampering.

8 By way of example, the above stated needs and others are met by a method
9 that includes providing an initial digital good to at least one computer. The initial
10 digital good is converted into a modified digital good using unique key data to
11 selectively manipulate at least one flow control operation within the initial digital
12 good, such that the modified digital good is operatively different in configuration,
13 but substantially functionally equivalent to the initial digital good.

14 The unique key data can be based on at least one unique identifier data
15 associated with a destination computer. For example, a source computer can
16 cryptographically generate the unique key data based on the unique identifier data
17 provided by the destination computer and a secret encryption key. The method
18 can include selectively limiting operation of the modified digital good to
19 computers that are properly associated with at least the unique identifier data
20 and/or unique key data.

21 The method can also include dividing the initial digital good into at least a
22 first portion and a second portion using the source computer. The first portion is
23 provided to the destination computer via a first computer readable medium, and a
24 modified second portion to the destination computer via a second computer
25 readable medium. Thus, for example, the first computer readable medium may

1 include a fixed computer readable medium, while the second computer readable
2 medium may include a network communication. The first portion is manipulated
3 or modified by the destination computer using a first key. Similarly, the source
4 computer manipulates the second portion using a second key.

5 When the initial digital good has been split into first and second portions,
6 then the modified digital good would therefore include a combination of the
7 modified first portion and the modified second portion. Since these portions were
8 operatively reconfigured using related keys/techniques, the modifications made to
9 each portion can be selected to match the modifications in the other.

10 Another aspect that is described herein is an arrangement that includes an
11 identifier configured to output unique identifier data associated with a computer,
12 and a key generator that is coupled to receive the unique identifier data and
13 generate at least one unique key data based on the received unique identifier data.
14 The arrangement also includes at least one individualizer that is configured to
15 receive the unique key data and at least a portion of an initial digital good, and
16 output at least a portion of a modified digital good using the unique key data to
17 selectively alter the initial digital good. Consequently, the modified digital good
18 will be operatively different in configuration, but substantially functionally
19 equivalent to the initial digital good.

DETAILED DESCRIPTION

Fig. 1 is a block diagram depicting an exemplary computer network 20 that is suitable for use with the various methods and arrangements in accordance with the present invention.

Computer network 20 includes a plurality of host or customer computers 22 coupled to at least one communications network 24. Communication network 24 is further coupled to at least one source or digital good provider computer 26. Computers 22 and 26 are configured to communicate with each other over communications network 24. By way of example, communications network 24 can include a public network such as the Internet. Communications network 24 might also include local-area networks, private wide-area networks, direct dial-up links, and the like.

In the discussion below, certain aspects of the present invention will be described in the general context of computer-executable instructions, such as program modules, being executed by one or more conventional personal computers. Generally, program modules include routines, programs, program segments, objects, components, data structures, etc. that perform particular tasks or implement particular abstract data types. Moreover, those skilled in the art will appreciate that the invention may be practiced with other computer system configurations, including hand-held devices, multiprocessor systems, microprocessor-based or programmable consumer electronics, network PCs, minicomputers, mainframe computers, and the like. In a distributed computer environment, program modules may be located in both local and remote memory storage devices.

1 Fig. 2 is a block diagram depicting a computer 102 that can be included in
2 customer computer 22 and/or provider computer 26, for example. Computer 102
3 includes one or more processors or processing units 104, a system memory 106,
4 and a bus 108 that couples various system components including the system
5 memory 106 to processors 104.

6 Bus 108 represents one or more of any of several types of bus structures,
7 including a memory bus or memory controller, a peripheral bus, an accelerated
8 graphics port, and a processor or local bus using any of a variety of bus
9 architectures. The system memory includes read only memory (ROM) 110 and
10 random access memory (RAM) 112. A basic input/output system (BIOS) 114,
11 containing the basic routines that help to transfer information between elements
12 within computer 102, such as during start-up, is stored in ROM 110. Computer
13 102 further includes a hard disk drive 116 for reading from and writing to a hard
14 disk, not shown, a magnetic disk drive 118 for reading from and writing to a
15 removable magnetic disk 120, and an optical disk drive 122 for reading from or
16 writing to a removable optical disk 124 such as a CD ROM, DVD ROM or other
17 optical media. The hard disk drive 116, magnetic disk drive 118, and optical disk
18 drive 122 are connected to the bus 108 by an SCSI interface 126 or some other
19 appropriate interface. The drives and their associated computer-readable media
20 provide nonvolatile storage of computer readable instructions, data structures,
21 program modules and other data for computer 102. Although the exemplary
22 environment described herein employs a hard disk, a removable magnetic disk 120
23 and a removable optical disk 124, it should be appreciated by those skilled in the
24 art that other types of computer readable media which can store data that is
25 accessible by a computer, such as magnetic cassettes, flash memory cards, digital

1 video disks, random access memories (RAMs) read only memories (ROM), and
2 the like, may also be used in the exemplary operating environment.

3 A number of program modules may be stored on the hard disk, magnetic
4 disk 120, optical disk 124, ROM 110, or RAM 112, including an operating system
5 130, one or more application programs 132, other program modules 134, and
6 program data 136. A user may enter commands and information into computer
7 102 through input devices such as keyboard 138 and pointing device 140. Other
8 input devices (not shown) may include a microphone, joystick, game pad, satellite
9 dish, scanner, or the like. These and other input devices are connected to the
10 processing unit 104 through an interface 142 that is coupled to the bus 108. A
11 monitor 144 or other type of display device is also connected to the bus 108 via an
12 interface, such as a video adapter 146. In addition to the monitor, personal
13 computers typically include other peripheral output devices (not shown) such as
14 speakers and printers.

15 Computer 102 can operate in a networked environment using logical
16 connections to one or more remote computers, such as a remote computer 148.
17 Remote computer 148 may be another personal computer, a server, a router, a
18 network PC, a peer device or other common network node, and typically includes
19 many or all of the elements described above relative to computer 102, although
20 only a memory storage device 150 has been illustrated in Fig. 2. The logical
21 connections depicted in Fig. 2 include a local area network (LAN) 152 and a wide
22 area network (WAN) 154. Such networking environments are commonplace in
23 offices, enterprise-wide computer networks, intranets, and the Internet.

24 When used in a LAN networking environment, computer 102 is connected
25 to the local network 152 through a network interface or adapter 156. When used

Generally, the data processors of computer 102 are programmed by means of instructions stored at different times in the various computer-readable storage media of the computer. Programs and operating systems are typically distributed, for example, on floppy disks or CD-ROMs. From there, they are installed or loaded into the secondary memory of a computer. At execution, they are loaded at least partially into the computer's primary electronic memory. The invention described herein includes these and other various types of computer-readable media when such media contain instructions or programs for implementing the steps described below in conjunction with a microprocessor or other data processor. The invention also includes the computer itself when programmed according to the methods and techniques described below. Furthermore, certain sub-components of the computer may be programmed to perform the functions and steps described below. The invention includes such sub-components when they are programmed as described. In addition, the invention described herein includes data structures, described below, as embodied on various types of memory media.

1 For purposes of illustration, software programs and other executable
2 program components such as the operating system are illustrated herein as discrete
3 blocks, although it is recognized that such programs and components reside at
4 various times in different storage components of the computer, and are executed
5 by the data processor(s) of the computer.

6 Reference is now made to Fig. 3, which is a block diagram depicting an
7 exemplary arrangement 200 that includes consumer computer 22 and provider
8 computer 26 and is configured to distribute and/or otherwise provide digital goods
9 to consumer computer 22 in a BORE-resistant manner. Here, a digital good "P"
10 202 is initially arranged within provider computer 26. Digital good P 202 can
11 include one or more computer programs, applications, operating systems, various
12 modules, functions, and/or content (e.g., music, video, books, etc.) and/or other
13 types of digital data, for example. Provider computer 26 is tasked to provide
14 digital good P 202 or an equivalent form thereof to consumer computer 22, such
15 that the resulting digital good on consumer computer 22 will be significantly
16 BORE resistant.

17 This is accomplished, in this example, by arranging provider computer 26
18 to deliver digital good P 202 in at least two stages. In a first stage, a first portion
19 "P1" 206 of digital good P 202 is delivered to consumer computer 22, for
20 example, via a CD ROM, DVD ROM, removable magnetic disk, preloaded on a
21 hard disk drive, solid-state memory device, a network connection, other
22 conventional computer readable media, or the like. In a second stage, a second
23 portion "P2" 207 of digital good P 202 (e.g., $P = P1 + P2$) is converted to a
24 modified second portion "Q2" based on identifying information provided by
25 consumer computer 22. The modified second portion Q2 is provided to consumer

1 computer 22. While modified second portion Q2 can be provided to consumer
2 computer 22 via any traditional/conventional computer readable medium, in this
3 example, modified second portion Q2 is provided to consumer computer 22 via a
4 network connection that allows for timely delivery.

5 Consumer computer 22, having received first portion P1 206, converts first
6 portion P1 206 to a modified first portion "Q1" using information provided by
7 provider computer 26. Consumer computer 22 is then able to combine modified
8 first portion Q1 with modified portion Q2 to produce a uniquely configured
9 modified digital good "Q" 218 (e.g., $Q = Q1 + Q2$) that is functionally equivalent
10 to digital good P 202.

11 With this basic process in mind, referring to Fig. 3, in this exemplary
12 arrangement digital good P 202 is split or otherwise divided into at least two
13 portions, e.g., P1 and P2, by a splitter 204. First portion P1 206 is provided to an
14 individualizer 208 within consumer computer 22. Second portion P2 207 is
15 provided to an individualizer 214 within provider computer 26. . By way of
16 example, individualizers 208 and 214 may include a program flow manipulator or
17 other like mechanism that allows the respective portions of digital good P 202 to
18 be operatively, functionally, sequentially, associatively, or otherwise individualized
19 based at least in part on one or more inputs. Here, for example, keys K1 and K2
20 are generated and/or otherwise provided to their respective individualizers 208 and
21 214 and used to "individualize" portions P1 and P2, respectively.

22 An identifier 210 within consumer computer 22, which may be
23 implemented in hardware and/or software, is essentially configured to uniquely
24 identify consumer computer 22 in some manner. By way of example, identifier
25 210 can include circuitry and/or functions that output unique identifying data

1 associated with processing unit 104, operating system 130, application programs
2 132, other modules 134, program data 136, other resources/subsystems within
3 computer 102, or coupled therewith. Identifier 210 may include information
4 associated with the consumer. For example, client identifier 210 might include
5 name, address, telephone, credit card, and/or other similar data. This and other
6 identifying information may be provided by one or more (optional) external
7 sources 211 to identifier 210 and/or provider computer 26. For example, external
8 sources 211 may include one or more computers, databases, human operators, etc.,
9 which provide the requisite identifying information to arrangement 200.

10 As shown, in this example the data output from client identifier 210 and/or
11 (optional) external sources 211 is provided to a key generator 212 within provider
12 computer 26. Key generator 212 is configured to generate one or more
13 cryptographically related encryption keys based at least in part on the identifying
14 information/data from client identifier 210 and/or external sources 211. Here, key
15 generator 212 generates two keys K1 and K2, which are cryptographically related
16 to a secret key K and at least a portion of the data from client identifier 210.
17 Consequently, keys K1 and K2 include data that is uniquely associated with
18 consumer computer 22 and/or the consumer associated therewith. Conventional
19 data encryption techniques are employed to insure that keys K1 and K2 cannot be
20 easily determined without access to secret key K. Once generated, key K1 is
21 provided to individualizer 208 within consumer computer 22, and key K2 is
22 provided to individualizer 214 within provider computer 26.

23 Individualizer 208, having received key K1, selectively individualizes first
24 portion P1 based on key K1. When a program flow manipulator is employed, for
25 example, this can include rearranging at least one program section, block of code,

004730-031400

1 pointer, address, adding/deleting code, etc., as definable within a program flow-
2 graph associated with first portion P1. Preferably, several modifications occur
3 within individualizer 208 to cause the resulting modified first portion Q1 to be
4 uniquely associated with key K1 and distinctly different from first portion P1 206.
5 Data from key K1 may be included within modified portion Q1. Modified first
6 portion Q1 is then provided to a combiner 216.

7 Similarly, individualizer 214, having received key K2, selectively
8 individualizes second portion P2 based on key K2. Again, when a program flow
9 manipulator is employed, for example this can include rearranging at least one
10 program section, block of code, pointer, address, adding/deleting code, etc., as
11 definable within a program flow-graph associated with second portion P2.
12 Preferably, several modifications occur within individualizer 214 to cause the
13 resulting modified second portion Q2 to be uniquely associated with key K2.
14 Modified second portion Q2 is then provided to combiner 216 within consumer
15 computer 22.

16 Combiner 216 is configured to combine modified first portion Q1 and
17 modified second portion Q2 to produce a modified digital good Q 218. Modified
18 digital good Q 218 is operatively configured to run within consumer computer 22.
19 Modified digital good Q 218 can be further configured to verify that information
20 from client identifier 210 matches related information, for example, data
21 associated with key K1, as incorporated in modified digital good Q 218. Thus,
22 modified digital good Q 218 can be designed to verify that the host computer that
23 it is running on, or attempting to be run on, is indeed authorized to do so.

24 In this manner, arrangement 200 causes the resulting configuration of
25 modified digital good Q 218 to be substantially unique for each particular

1 computer and/or consumer. Arrangement 200 is significantly BORE resistant,
2 since the security features of each unique implementation of modified digital good
3 Q 218 are inherently unique and would require potential hackers to expend a great
4 deal of effort to discover, override and/or otherwise disable the features. Thus,
5 rather than posing a "break once" situation, the present invention would require
6 hackers to "break each" modified digital good Q 218.

7 Additional security features can also be included or otherwise incorporated
8 in modified digital good Q 218, such as, for example, various encryption, data
9 hiding and/or fingerprinting techniques can be employed to further discourage
10 unauthorized use or distribution. Thus, with respect to Fig. 3, for example, digital
11 good P 202 can be further pre-processed prior to being provided to splitter 204.
12 Portions P1 206 and/or P2 207 can be further post-processed prior to being
13 supplied to individualizers 208 and 214, respectively. Similarly, additional
14 pre/post-processing can be conducted on modified first portions Q1 and/or Q2.
15 Such security features may include local data such as, for example, time and date,
16 serial numbers, random numbers, other public/private keys, digital certificates,
17 digital signatures, etc. In certain configurations, provider computer 26 may also
18 store certain types of information in a local database (not shown).

19 Those skilled in the art will recognize that the processing described above
20 can be selectively distributed and/or scheduled as needed. Indeed, in certain
21 arrangements, processes that are computationally intensive may be completed
22 offline or on other computers (not shown). Thus, for example, if individualizer
23 208 includes a program flow manipulator, it may be prudent to run the program
24 flow manipulator on another computer rather than tie up consumer computer 22.

25

1 In other arrangements, splitter 204 may also be provided through one or more
2 other computers.

3 In accordance with certain further aspects, arrangement 200 of Fig. 3 can
4 even be employed when either first portion P1 206 or second portion P2 207
5 contains no data (i.e., $P1=P$, or $P2=P$).

6 Exemplary implementations in such cases are depicted in Figs 4 and 5, as
7 described below. Basically, if either first portion P1 206 or second portion P2 207
8 contains no data, then certain functionality within arrangement 200 of Fig. 3 can
9 be eliminated or otherwise ignored.

10 Fig. 4 is a block diagram depicting another exemplary arrangement 220, in
11 accordance with certain further aspects of the present invention. As shown, in this
12 example, digital good P 202 is not split into portions. Instead, digital good P 202
13 is provided to individualizer 208. Key generator 212 is configured to generate key
14 K1 based on data from identifier 210. Key K1 is then provided to individualizer
15 208. Individualizer 208 converts digital good P 202 into modified digital good Q1
16 218.

17 Fig. 5 is a block diagram depicting yet another exemplary arrangement 230.
18 As shown, in this example, digital good P 202 is not split into portions. Instead,
19 digital good P 202 is provided to individualizer 214. Key generator 212 generates
20 key K2 based on data from identifier 210. Key K2 is provided to individualizer
21 214. Individualizer 214 then converts digital good P 202 into a modified digital
22 good Q2 218. Modified digital good Q2 218 is then provided to consumer
23 computer 22.

24 Fig. 6 is a block diagram that illustratively depicts certain exemplary
25 features of a BORE-resistant digital good as configured and distributed, for

1 example, by arrangement 200 in Fig. 3, as described above. In this example,
2 digital good P 202 includes a plurality of segments or blocks 240 that are
3 operatively or associatively configured together in some manner, for example, as
4 represented by the interconnecting arrows between various blocks. Thus, for
5 example, the arrow between “block A” and “block B” can represent a calling
6 function, a pointer, data passing, a content sequence, a content ordering, or the
7 like.

8 As shown, digital good P 202 has been selectively split into a first portion
9 P1 206 and second portion P2 207. Here, first portion P1 206 includes “block A”,
10 “block B”, “block C”, “block D”, and “block G”. Second portion P2 207 includes
11 “block E”, “block F”, “block H”, and “block I”.

12 As a result of arrangement 200, in Fig. 3, for example, a modified digital
13 good Q 218 has been created as shown at the bottom of Fig. 6. Here, the blocks
14 240 have been rearranged as blocks 242, and operatively or associatively
15 reconfigured as represented, for example, by arrows 244a-c. This produces a
16 functionally equivalent version of digital good P 202. Thus, for example, arrow
17 244a illustrates that “block I” and “block G” are now operatively or associatively
18 coupled, arrow 244b illustrates that “block F” and “block H” are now operatively
19 or associatively coupled, and arrow 244c illustrates that “block H” and “block D”
20 are now operatively or associatively coupled, where they were not previously.
21 Similarly, the absence of an arrow between “block A” and “block B” represents
22 that they are no longer directly operatively or associatively coupled as before, but
23 rather “block C” has been introduced there between.

24 Those skilled in the art will recognize that a variety of different
25 permutations are available in configuring digital good P 202 into corresponding

1 modified digital good Q 218, and that certain configurations will be more optimal
2 than others. For this reason and others, splitter 204, individualizers 208 and 214,
3 and/or combiner 216 can be further arranged to configure digital good Q 218 to
4 meet certain performance goals, as well as DRM goals.

5 Fig. 7 is a flow-chart depicting an exemplary process 300 for providing a
6 BORE-resistant digital good to a computer 102, as in Fig. 2, for example, using
7 arrangement 200. In step 302, the digital good provider (e.g., a vendor) supplies a
8 first portion P1 206 of a digital good P 202 to a consumer. In step 304, the
9 consumer supplies requisite identifying information to the vendor. In step 304, the
10 vendor may also or optionally access identifying information within additional
11 external resources. Next, in step 306, the vendor generates cryptographically
12 related keys K1 and K2 based at least in part on the identifying information in step
13 304.

14 In step 308, the vendor individualizes at least part of a second portion P2 of
15 digital good P 202, using key K2. This results in a modified second portion Q2.
16 The vendor provides modified second portion Q2 and key K1 to the consumer.

17 In step 310, the consumer individualizes first portion P1 206 using key K1,
18 which results in a modified portion Q1. Next, in step 312, the consumer combines
19 modified first portion Q1 and modified second portion Q2 to produce a modified
20 digital good Q 218, which is uniquely and operatively associated with the
21 consumer and substantially functionally equivalent to digital good P 202.

22 Fig. 8 is a flow-chart depicting an exemplary process 400 for configuring a
23 digital good using the BORE-resistant techniques as described above. In this
24 example, the digital good is assumed to be a software program. In step 402, a first
25 plurality of program segments associated with digital good P 202 are provided. In

1 step 404, unique key data associated with an identifiable computer/consumer is
2 provided. Next, as shown in step 406, at least a portion of a program flow within
3 the first plurality of segments is modified based on the unique key data. In step
4 408, a unique digital good is provided for use by the identifiable
5 computer/consumer, using at least the modified first plurality of segments from
6 step 406.

7 Fig. 9 is a flow-chart depicting an exemplary process 420 for operating a
8 computer 102, as in Fig. 2, for example, using a BORE-resistant digital good that
9 has been configured using the BORE-resistant techniques as described above.
10 Here, in step 422, a uniquely configured digital good is provided for use by an
11 identifiable computer/consumer. In step 424, unique key data associated with the
12 identifiable computer/consumer is also provided. Next, in step 426, the uniquely
13 configured digital good is selectively verified, using the unique key data, as being
14 properly associated with an identifiable computer/consumer running or attempting
15 to run the unique configuration digital good. The uniquely configured digital good
16 will be unable to properly/fully function, or to be otherwise fully accessed, if the
17 identifiable computer/consumer cannot be properly verified in step 426.

18 The preceding exemplary methods and arrangements may be implemented
19 in an automated and controlled manner, such that neither the consumer nor the
20 digital good provider is overly burdened.

21 Although the invention has been described in language specific to structural
22 features and/or methodological steps, it is to be understood that the invention
23 defined in the appended claims is not necessarily limited to the specific features or
24 steps described. Rather, the specific features and steps are disclosed as preferred
25 forms of implementing the claimed invention.